UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/919,185	07/30/2001	Edward B. Boden	END920010019US1	2635

7590 11/25/2005

IBM Corporation
Intellectual Property Law (Dept. 917, Bldg. 006-1)
3605 Highway 52 North
Rochester, MN 55901-7829

EXAMINER	
LESNIEWSKI, VICTOR D	

ART UNIT	PAPER NUMBER
2152	

DATE MAILED: 11/25/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/919,185	BODEN, EDWARD B.	
	Examiner	Art Unit	
	Victor Lesniewski	2152	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *09 September 2005*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-5,8-14,18-25,29,30,34-41,43 and 45-53* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-5,8-14,18-25,29,30,34-41,43 and 45-53* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. The amendment filed 9/9/2005 has been placed of record in the file.

2. Claims 1, 10, 18, 22-25, 29, 30, 34, 36-41, 43, and 45-53 have been amended.

3. Claims 6, 7, 15-17, 26-28, 31-33, 42, and 44 have been canceled.

4. The rejection of the claims under 35 U.S.C. 101 is withdrawn in view of the amendment.

5. Claims 1-5, 8-14, 18-25, 29, 30, 34-41, 43, and 45-53 are now pending.

6. The applicant's arguments with respect to claims 1-5, 8-14, 18-25, 29, 30, 34-41, 43, and 45-53 have been considered but are moot in view of the following new grounds of rejection.

Response to Amendment

7. Claims have been amended to further recite details of the look-ahead function and of the filter processing. The amendment proves a change in scope to the independent claims as the independent claims now explicitly state greater detail concerning the look-ahead function or the execution of the filter processing. However, none of the amended claims show a patentable distinction over the prior art as evidenced by the following new grounds of rejection.

8. Several status identifiers in the amendment have been found to be improper. Please refer to 37 CFR 1.21(c) and submit the proper status identifiers in any future amendments.

Claim Rejections - 35 USC § 103

9. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person

having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

10. Claims 1-5, 8-14, 22, 24, 25, 29, 30, 34-37, 39, 41, 43, 45, 47-50, 52, and 53 are rejected under 35 U.S.C. 103(a) as being unpatentable over Lucovsky (U.S. Patent Number 6,868,450) in view of Jackowski et al. (U.S. Patent Number 6,141,686), hereinafter referred to as Jackowski.

11. Lucovsky disclosed a system for packet filtering based on a process attribute. In an analogous art, Jackowski disclosed a system for packet classification based on high-level applications.

12. Concerning claims 1, 10, 22, 25, 30, 34, 36, 37, 39, 41, 43, 45, 48, 49, and 53, Lucovsky did not explicitly state a look-ahead function being executed within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for said inbound packet, said IP layer provides to said transport layer said inbound packet and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered. However, Jackowski does explicitly disclose this feature as his system uses an extensible service provider within the stack to identify high-level applications. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify the system of Lucovsky by adding the ability to utilize a look-ahead function being executed within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for said inbound packet, said IP layer provides to said transport layer said inbound packet and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered as provided by Jackowski. Here the combination satisfies the need for a system that

prioritizes network traffic based on high-level applications and users rather than low-level IP addresses and TCP ports. See Jackowski, column 4, lines 38-48. This rationale also applies to those dependent claims utilizing the same combination.

13. Concerning claims 1, 10, 22, 24, 29, 34, 36, 37, 39, 41, 43, 45, 47, 49, 50, and 52, Lucovsky did not explicitly state filter processing including constructing and evaluating logical expressions of arbitrary length, and selectively using a set of logical operators, alternative filter selector fields, and value set. However, such filter constructions are well known in the art as evidenced by Jackowski whose system uses dynamic policy rules in order to classify packets. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify the system of Lucovsky by adding the ability to utilize filter processing including constructing and evaluating logical expressions of arbitrary length, and selectively using a set of logical operators, alternative filter selector fields, and value set as provided by Jackowski. Again the combination satisfies the need for a system that prioritizes network traffic based on high-level applications and users rather than low-level IP addresses and TCP ports. See Jackowski, column 4, lines 38-48. This rationale also applies to those dependent claims utilizing the same combination.

14. Concerning claims 1, 10, 22, 25, 30, 34, 36, 37, 39, 41, 43, 45, 48, 49, and 53, Lucovsky does not explicitly state marking a packet as non-deliverable. However, taking some action on a packet before passing it on for further filtering is well known in the art and Lucovsky has disclosed a variety of features that could be used for such an action. In one instance Lucovsky discusses values that uniquely identify each process. See column 5, lines 18-24. For example, a packet could be marked as non-deliverable before being passed to the sockets layer if its data

does not correctly identify a certain value of a certain process. Also, authentication or authorization techniques are well known in the art that may lead to a pre-assessment of a packet before it is passed to the sockets layer. Thus, it would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify the system of Lucovsky by adding the ability to mark a packet as non-deliverable.

15. Thereby, the combination of Lucovsky and Jackowski discloses:

- <Claim 1>

  A method for control and management of communication traffic, comprising the steps of: expressing access rules as filters referencing system kernel data (Lucovsky, column 2, lines 13-35); for outbound processing, determining source application indicia (Lucovsky, column 7, lines 51-65); for inbound packet processing, executing a look-ahead function to determine target application indicia; said look-ahead function being executed within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for said inbound packet, said IP layer provides to said transport layer said inbound packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32); and responsive to said source or target application indicia, executing filter processing; said filter processing including constructing and evaluating logical expressions of arbitrary length, and selectively using a set of logical operators, alternative filter selector fields, and value set (Lucovsky, column 8, lines 11-16 and 33-40 and Jackowski, column

15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7).

- <Claim 2>

  The method of claim 1, further comprising the steps of executing said determining and executing steps within a kernel filtering function upon encountering a filter selector field referencing kernel data not included in said packet (Lucovsky, column 8, lines 17-22).

- <Claim 3>

  The method of claim 1, said filter processing including the steps of: determining a task or thread identifier (Lucovsky, figure 2, items 101 and 102); based on said task or thread identifier, determining a process or job identifier (Lucovsky, column 4, lines 53-59); and based on said process or job identifier, determining job or process attributes for filter processing (Lucovsky, column 7, lines 6-11 and figure 2, items 119 and 120).

- <Claim 4>

  The method of claim 1, said filter processing including the steps of: determining a user identifier (Lucovsky, column 4, lines 53-59); and based on said user identifier, determining user attributes for filter processing (Lucovsky, column 7, lines 6-11 and figure 2, items 119 and 120).

- <Claim 5>

  The method of claim 3, further comprising the step of determining from said task identifier a work control block containing said process or job identifier (Jackowski, column 12, lines 15-29).

- <Claim 8>

  The method of claim 1, further comprising the steps of: delivering to said filters infrastructure access rules for defining security context (Lucovsky, column 1, lines 44-58 and column 2, lines 36-48).

- <Claim 9>

  The method of claim 8, said infrastructure including logging, auditing, and filter rule load controls (Jackowski, column 11, line 41 through column 12, line 14).

- <Claim 10>

  A method for control and management of aspects of communication traffic within filtering, comprising the steps of: receiving IP packet data into a TCP/IP protocol stack executing within a system kernel (Lucovsky, column 2, lines 13-35); for inbound packet processing, executing a look-ahead function within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for said IP inbound packet, said IP layer provides to said transport layer said inbound IP packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32); and executing filtering code within said system kernel with respect to non-IP packet data accessed within said system kernel outside of said TCP/IP protocol stack (Lucovsky, column 8, lines 11-16 and 33-40); said filtering code constructing and evaluating logical expressions of arbitrary length, and selectively using a set of logical operators, alternative filter selector fields,

and value set (Jackowski, column 15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7).

- <Claim 11>

  The method of claim 10, said non-IP packet data including context data regarding said IP packet (Lucovsky, column 8, lines 23-32).

- <Claim 12>

  The method claim 10, said non-IP packet data including data specific to a task generating said non-IP packet data (Lucovsky, column 7, lines 51-65).

- <Claim 13>

  The method of claim 10, said non-IP packet data including data specific to a task that will receive said IP packet (Lucovsky, column 8, lines 23-32).

- <Claim 14>

  The method of claim 11, said context data including packet arrival interface indicia (Lucovsky, column 8, lines 23-32).

- <Claim 22>

  A method for traversing a portion only of a protocol stack to disallow selective IP packet traffic, comprising the steps of: receiving a packet in the kernel of the operating system of a first node from an application, said kernel including filter processor (Lucovsky, column 2, lines 13-35); said filter processor for constructing and evaluating logical expressions of arbitrary length, said logical expressions selectively including a set of logical operators, alternative filter selector fields, and value set (Jackowski, column 15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7); for inbound packet

processing to a first node from a second node, executing a look-ahead function in the system kernel of said first node to determine a target application; said system kernel including a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for said inbound packet, said IP layer provides to said transport layer said inbound packet, marked as non-deliverable, and receives back from said transport layer indicia identifying the application layer application to which said packet would have been delivered (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32); for both said inbound packet processing, and for outbound packet processing from said first node to said second node, executing within said kernel the steps of processing said packet by determining a task ID (Lucovsky, figure 2, items 101 and 102); responsive to said task ID, determining a corresponding work control block (Jackowski, column 12, lines 15-29); determining a user process or job identifier from said work control block (Lucovsky, column 4, lines 53-59 and Jackowski, column 12, lines 15-29)); from the user process or job identifier selectively determining attributes for said user process or job (Lucovsky, column 7, lines 6-11 and figure 2, items 119 and 120); and passing said attributes to said filter processor for managing and controlling communication traffic (Lucovsky, column 8, lines 11-16 and 33-40).

- <Claim 24>

  A method for managing and controlling communication traffic by centralizing access rules in filters executing within and referencing data available in system kernels, comprising the steps for outbound packet processing from a first node to a second node

of: receiving said packet in the kernel of the operating system of said first node from an application or process at said first node (Lucovsky, column 7, lines 51-65); processing said packet by determining a task ID (Lucovsky, figure 2, items 101 and 102); responsive to said task ID, determining a corresponding work control block (Jackowski, column 12, lines 15-29); responsive to said work control block, determining a process or job identifier (Lucovsky, column 4, lines 53-59); responsive to said process job identifier, determining job or process attributes (Lucovsky, column 7, lines 6-11 figure 2, items 119 and 120).

- <Claim 25>

  The method of claim 24, further comprising the steps for inbound packet processing from said second node to said first node of: initially operating said kernel at said first node to determine a target application for said packet at said first node by executing a look-ahead function within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for said inbound packet, said IP layer provides to said transport layer said inbound packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32).

- <Claim 29>

  A method for managing and controlling communication traffic by centralizing the access rules, comprising the steps for outbound packet processing from a first node to a second

node of: receiving said packet in the kernel of the operating system of said first node from an application or process at said first node, said kernel including a filter processor for constructing and evaluating logical expressions of arbitrary length, said logical expressions selectively including a set of logical operators, alternative filter selector fields, and value set (Lucovsky, column 7, lines 51-65 and Jackowski, column 15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7); processing said packet by determining a task ID (Lucovsky, figure 2, items 101 and 102); responsive to said task ID, determining a corresponding work control block (Jackowski, column 12, lines 15-29); determining a user ID control block from said work control block (Lucovsky, column 4, lines 53-59); from the user ID control block determining attributes for said user (Lucosky, column 7, lines 6-11 and figure 2, items 119 and 120); and passing said attributes to said filter processor for managing and controlling communication traffic (Lucovsky, column 8, lines 11-16 and 33-40).

- <Claim 30>

The method of claim 29, further comprising the steps for inbound packet processing from said second node to said first node of: initially operating said kernel at said first node to determine a target application for said packet at said first node by executing a look-ahead function within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for said inbound packet, said IP layer provides to said transport layer said inbound packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been

delivered (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32).

- <Claim 34>

  A method for control and management of communication traffic with respect to a system node, comprising the steps of: receiving at said system node an inbound packet (Lucovsky, column 8, lines 23-32); and executing within a protocol stack of the system kernel of said system node a filtering function identifying for said inbound packet a filter referencing non-packet data, and constructing and evaluating logical expressions of arbitrary length, said logical expressions selectively including a set of logical operators, alternative filter selector fields, and value set (Lucovsky, column 2, lines 13-35 and Jackowski, column 15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7); and responsive to said filter, executing a look-ahead function for identifying a target application for said inbound packet; said look-ahead function executed within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for said IP inbound packet, said IP layer provides to said transport layer said inbound packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32).

- <Claim 35>

  The look-ahead function of the method of claim 34 further comprising the steps of: passing to a transport layer function identified by an IP header a packet marked non-deliverable for determining which user-level process or job is to receive said packet (Jackowski, column 12, lines 15-29 and obviousness as discussed above); receiving from said transport layer an application layer task identifier said user-level process or job (Lucovsky, column 8, lines 23-32 and figure 2, items 101 and 102); and thereafter passing said packet marked by said task identifier to said transport layer for delivery to said application layer task (Lucovsky, column 8, lines 33-40).

- <Claim 36>

  System for control and management of communication traffic, comprising: a system kernel including a filter function and stack data (Lucovsky, column 2, lines 13-35); said filter function including a filter selectively referencing said stack data for expressing access rules (Lucovsky, column 2, lines 13-35); said filter function being responsive to receipt of an outbound packet determining a source application (Lucovsky, column 7, lines 51-65); said filter function being responsive to receipt of an inbound packet processing for executing a look-ahead function within a protocol stack to determine a target application; said protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for said inbound packet, said IP layer provides to said transport layer said inbound packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been

delivered (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32); and said filter function being responsive to said source or target application for executing filter processing including constructing and evaluating logical expressions of arbitrary length, said logical expressions selectively including a set of logical operators, alternative filter selector fields, and value set (Lucovsky, column 8, lines 11-16 and 33-40 and Jackowski, column 15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7).

- <Claim 37>

  A system for control and management of aspects of communication traffic within filtering, comprising: a system kernel (Lucovsky, column 2, lines 13-35); a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer for executing within said system kernel, responsive to an inbound IP packet, a look-ahead function by which said IP layer provides to said transport layer said inbound IP packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32); and filtering code within said system kernel operable with respect to non-IP packet data accessed within said system kernel outside of said protocol stack for controlling and managing said aspects of communication traffic (Lucovsky, column 2, lines 13-35); said filtering code for constructing and evaluating logical expressions of arbitrary length, said logical expressions selectively including a set of logical operators, alternative filter selector

fields, and value set (Jackowski, column 15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7).

- <Claim 39>

  A system for traversing a portion only of a protocol stack to disallow selective IP packet traffic, comprising: a system kernel (Lucovsky, column 2, lines 13-35); a filter processor executing within said system kernel for constructing and evaluating logical expressions of arbitrary length, said logical expressions selectively including a set of logical operators, alternative filter selector fields, and value set (Lucovsky, column 2, lines 13-35 and Jackowski, column 15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7); said filter processor responsive to an inbound packet for executing a look-ahead function for determining a target application; said look-ahead function operating within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for said IP inbound packet, said IP layer provides to said transport layer said inbound IP packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32); said filter processor responsive to both inbound and outbound packets for processing said packet by determining a task ID (Lucovsky, figure 2, items 101 and 102); responsive to said task ID, determining a corresponding work control block (Jackowski, column 12, lines 15-29); determining a user ID, process or job identifier from said work control block (Lucovsky, column 4, lines 53-59); from the user

ID, process or job identifier selectively determining attributes for said user process or job (Lucovsky, column 7, lines 6-11 and figure 2, items 119 and 120); and passing said attributes to said filter processor for managing and controlling communication traffic (Lucovsky, column 8, lines 11-16 and 33-40).

- <Claim 41>

A system for managing and controlling communication traffic by centralizing access rules in filters executing within and referencing data available in system kernels, comprising: a computer readable medium; first code for receiving a packet in the kernel of the operating system of a first node from an application or process at said first node; said kernel responsive to an inbound packet for executing a look-ahead function within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for said inbound packet, said IP layer provides to said transport layer said inbound IP packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32); second code for processing said packet by determining a task ID (Lucovsky, figure 2, items 101 and 102); third code responsive to said task ID for determining a corresponding work control block (Jackowski, column 12, lines 15-29); fourth code responsive to said work control block for determining a process or job identifier (Lucovsky, column 4, lines 53-59); fifth code responsive to said process or job identifier for determining job or process attributes (Lucovsky, column 7, lines 6-11 and figure 2, items 119 and 120); sixth

code for executing said filters by constructing and evaluating logical expressions of arbitrary length, said logical expressions selectively including a set of logical operators, alternative filter selector fields, and value set (Jackowski, column 15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7); and wherein said first, second, third, fourth, fifth, and sixth code is recorded on said computer readable medium.

- <Claim 43>

  A system for control and management of communication traffic with respect to a system node, comprising: a filtering function executing within a protocol stack of the system kernel of said system node identifying for an inbound packet a filter referencing non-packet data (Lucovsky, column 2, lines 13-35 and column 8, lines 23-32); and a look-ahead function responsive to said filter for identifying a target application for said inbound packet; said look-ahead function functioning within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for said inbound packet, said IP layer provides to said transport layer said inbound packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32); ; and a filter processor for constructing and evaluating logical expressions of arbitrary length, said logical expressions selectively including a set of logical operators, alternative filter selector

fields, and value set (Jackowski, column 15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7).

- <Claim 45>

  A computer program product for control and management of aspects of communication traffic within filtering, said computer program product comprising: a computer readable medium; first program instructions to receive IP packet data into a TCP/IP protocol stack executing within a system kernel (Lucovsky, column 2, lines 13-35) including, for processing an inbound IP packet, a look-ahead function within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for said IP inbound packet, said IP layer provides to said transport layer said inbound packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32); second program instructions to execute filtering code within said system kernel with respect to non-IP packet data accessed within said system kernel outside of said TCP/IP protocol stack (Lucovsky, column 8, lines 11-16 and 33-40) by constructing and evaluating logical expressions of arbitrary length, said logical expressions selectively including a set of logical operators, alternative filter selector fields, and value set (Jackowski, column 15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7); and wherein said first and second program instructions are recorded on said medium.

- <Claim 47>

  A computer program product for managing and controlling communication traffic by centralizing access rules in filters executing within and referencing data available in system kernels, said computer program product comprising: a computer readable medium; first program instructions to receive said packet in the kernel of the operating system of said first node from an application or process at said first node (Lucovsky, column 7, lines 51-65); second program instructions to process said packet by determining a task ID (Lucovsky, figure 2, items 101 and 102); third program instructions, responsive to said task ID, determining a corresponding work control block (Jackowski, column 12, lines 15-29); fourth program instructions, responsive to said work control block, to determine a process or job identifier (Lucovsky, column 4, lines 53-59); fifth program instructions, responsive to said process or job identifier, to determine job or process attributes (Lucovsky, column 7, lines 6-11 and figure 2, items 119 and 120); and sixth program instructions to execute a filter processor for constructing and evaluating logical expressions of arbitrary length, said logical expressions selectively including a set of logical operators, alternative filter selector fields, and value set (Jackowski, column 15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7); and wherein said first, second, third, fourth, fifth, and sixth program instructions are recorded on said medium.

- <Claim 48>

  The computer program product of claim 47, said computer program product further comprising for inbound packet processing from said second node to said first node: sixth

program instructions to initially operate said kernel at said first node to determine a target application for said packet at said first node by executing a look-ahead function within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for said IP inbound packet, said IP layer provides to said transport layer said inbound IP packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32); and wherein said sixth program instructions are recorded on said medium.

- <Claim 49>

  A computer program product for control and management of communication traffic, comprising: a computer readable medium; first program instructions for expressing access rules as filters referencing system kernel data (Lucovsky, column 2, lines 13-35); second program instructions, for outbound processing, for determining a source application (Lucovsky, column 7, lines 51-65); third program instructions, for inbound packet processing, for executing a look-ahead function to determine a target application; said look-ahead function operating within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for said IP inbound packet, said IP layer provides to said transport layer said inbound IP packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered (Lucovsky, column 8, lines 23-32 and

Jackowski, column 15, line 66 through column 16, line 32); fourth program instructions, selectively responsive to said source and target application, for executing filter processing including constructing and evaluating logical expressions of arbitrary length, said logical expressions selectively including a set of logical operators, alternative filter selector fields, and value set (Lucovsky, column 8, lines 11-16 and 33-40 and Jackowski, column 15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7); ; and wherein said first, second, third, and fourth program instructions are recorded on said computer readable medium.

- <Claim 50>

  A computer program product for control and management of aspects of communication traffic within filtering, comprising: a computer readable medium; first program instructions for receiving IP packet data into a TCP/IP protocol stack executing within a system kernel (Lucovsky, column 2, lines 13-35) second program instructions for executing filtering code within said system kernel with respect to non-IP packet data accessed within said system kernel outside of said TCP/IP protocol stack (Lucovsky, column 8, lines 11-16 and 33-40); said filtering code constructing and evaluating logical expressions of arbitrary length, said logical expressions selectively including a set of logical operators, alternative filter selector fields, and value set (Jackowski, column 15, lines 21-43; column 11, lines 41-46; column 12, line 61 through column 13, line 7); and wherein said first and second program instructions are recorded on said computer readable medium.

- <Claim 52>

  A computer program product for managing and controlling communication traffic by centralizing access rules in filters executing within, and referencing data available in, system kernels, comprising: a computer readable medium; first program instructions for receiving said packet in the kernel of the operating system of said first node from an application or process at said first node (Lucovsky, column 7, lines 51-65); second program instructions for processing said packet by determining a task ID (Lucovsky, figure 2, items 101 and 102); third program instructions, responsive to said task ID, for determining a corresponding work control block (Jackowski, column 12, lines 15-29); fourth program instructions, responsive to said work control block, for determining a process or job identifier (Lucovsky, column 4, lines 53-59); fifth program instructions, responsive to said process or job identifier, for determining job or process attributes (Lucovsky, column 7, lines 6-11 and figure 2, items 119 and 120); sixth program instructions for executing a filter processor for constructing and evaluating logical expressions of arbitrary length, said logical expressions selectively including a set of logical operators, alternative filter selector fields, and value set (Jackowski, column 15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7); and wherein said first, second, third, fourth, fifth, and sixth program instructions are recorded on said computer readable medium.

- <Claim 53>

  The computer program product of claim 52, further comprising for inbound packet processing from said second node to said first node: seventh program instructions initially

operating said kernel at said first node to determine a target application for said packet at said first node by executing a look-ahead function within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for said IP inbound packet, said IP layer provides to said transport layer said inbound IP packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32); ; and wherein said seventh program instructions are recorded on said computer readable medium.

Since the combination of Lucovsky and Jackowski discloses all of the above limitations, claims 1-5, 8-14, 22, 24, 25, 29, 30, 34-37, 39, 41, 43, 45, 47-50, 52, and 53 are rejected.

16. Claims 18-21, 23, 38, 40, 46, and 51 are rejected under 35 U.S.C. 103(a) as being unpatentable over Lucovsky in view of Jackowski, as applied above, further in view of Fiveash et al. (U.S. Patent Number 6,076,168), hereinafter referred to as Fiveash.

17. The combination of Lucovsky and Jackowski disclosed a system for packet filtering based on a process attribute and high-level application. In an analogous art, Fiveash disclosed a method for securing data traffic between host systems that uses a filter having rules associated with a defined tunnel.

18. Concerning claim 21, the combination of Lucovsky and Jackowski does not explicitly state establishing a tunnel between two IP addresses. However, Lucovsky does discuss processes that operate between two unique port numbers. Furthermore, Fiveash's system is

based on the use of a tunnel bound at each end to keep data confidential. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify the combination of Lucovsky and Jackowski by adding the ability to establish a tunnel between two IP addresses as provided by Fiveash. Here the combination satisfies the need for a filter mechanism that can determine whether a process having a certain attribute may access a network. See Lucovsky, column 2, lines 5-10.

19. Concerning claims 18, 23, 38, 40, 46, and 51, the combination of Lucovsky and Jackowski does not explicitly state the use of a filter statements syntax. Although Jackowski states the use of various parameters and values that are collected and utilized by his policy server, he does not set forth a specific syntax for analysis of the data. However, packet filtering systems that allow users to provide the parameters by using a filter statements syntax are well known in the art as evidenced by Fiveash. Fiveash states an exemplary list of rules that are used for filtering packets where the parameters of the rules are set by the user of the host system. See figure 4. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify the combination of Lucovsky and Jackowski by adding the ability to provide filter statements syntax as provided by Fiveash. Again the combination satisfies the need for a filter mechanism that can determine whether a process having a certain attribute may access a network. See Lucovsky, column 2, lines 5-10.

20. Thereby, the combination of Lucovsky, Jackowski, and Fiveash discloses:

- <Claim 18>

  A method for centralizing system-wide communication management and control within filter rules, comprising the steps of: providing filter statements syntax for accepting

parameters in the form of a selector, each selector specifying selector field, operator, and a set of values (Fiveash, figure 4); for an inbound packet, executing a look-ahead function within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for said inbound packet, said IP layer provides to said transport layer said inbound packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered by said sockets layer (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32); said selector referencing data that does not exist in IP packets (Lucovsky, column 2, lines 13-35); processing said filter statements, including contructing and evaluating logical expressions of arbitrary length, and selectively using a set of logical operators, alternative filter selector fields, and value set (Jackowski, column 15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7).

- <Claim 19>

  The method of claim 18, said parameters selectively including userid, user profile, user class, user group, user group authority, user special authority, job name, process name, job group, job class, job priority, other job or process attributes, and date & time (Lucovsky, column 4, lines 53-59).

- <Claim 20>

  The method of claim 18, said filters statements being provided within a user interface to said system (Fiveash, column 3, lines 57-58).

- <Claim 21>

  The method of claim 18, further comprising the steps of: establishing a tunnel between two IP address limiting traffic to applications bound to ports at each end of said tunnel (Lucovsky, column 5, lines 6-23 and Fiveash, column 3, lines 64-67); said filtering code accessing filtering attributes further limiting traffic selectively to job indicia (Lucovsky, column 4, lines 53-59 and column 7, lines 6-11); and operating said filtering code within a kernel filtering function upon encountering a filter selector field referencing kernel data not included in said traffic (Lucovsky, column 8, lines 11-16 and 33-40).

- <Claim 23>

  A method for expressing access rules as filters, comprising the steps of: providing a filter statements syntax for accepting parameters in the form of a selector, each selector specifying selector field, operator, and a set of values (Fiveash, figure 4); and said selector referencing data that does not exist in IP packets for controlling access to an application (Lucovsky, column 2, lines 13-35); for an inbound packet, executing a look-ahead function within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for said IP inbound packet, said IP layer provides to said transport layer said inbound IP packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32); and processing said filter statements by constructing and evaluating logical expressions of arbitrary length, said logical expressions selectively

including a set of logical operators, alternative filter selector fields, and value set referencing said application layer application (Jackowski, column 15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7).

- <Claim 38>

  A system for centralizing system-wide communication management and control within filter rules, comprising: filter statements having a syntax for accepting parameters in the form of a selector, each selector specifying selector field, operator, and a set of values (Fiveash, figure 4); said selector referencing data that does not exist in IP packets (Lucovsky, column 2, lines 13-35); a look-ahead function within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for an inbound packet, said IP layer provides to said transport layer said inbound packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32); and a filter processor for constructing and evaluating filter statements including logical expressions of arbitrary length, said logical expressions selectively including a set of logical operators, alternative filter selector fields, and value set (Jackowski, column 15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7).

- <Claim 40>

  A system for expressing access rules as filters, comprising: filter statements for accepting parameters in the form of a selector, each selector specifying selector field, operator, and

a set of values (Fiveash, figure 4); said selector referencing data that does not exist in IP packets controlling access to an application (Lucovsky, column 2, lines 13-35); a look-ahead executing within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for an inbound packet, said IP layer provides to said transport layer said inbound packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32); and a filter processor for constructing and evaluating said filter statements as logical expressions of arbitrary length, each said logical expression selectively including said operator selected from a set of logical operators, alternative filter selector fields, and value set (Jackowski, column 15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7).

- <Claim 46>

A computer program product for centralizing system-wide communication management and control within filter rules, said computer program product comprising: a computer readable medium; first program instructions to execute filter statements having a syntax for accepting parameters in the form of a selector, each selector specifying selector field, a logical operator selected from a set of a plurality of logical operators, and a set of values (Fiveash, figure 4 and Jackowski, column 15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7); and second program instructions to cause said selector to reference data that does not exist in IP packets (Lucovsky, column

2, lines 13-35), said data including application layer indicia obtained for an incoming packet by a look-ahead function, said look-ahead function executing within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and which, for said IP inbound packet, said IP layer provides to said transport layer said inbound IP packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32); and wherein said first and second program instruction are recorded on said medium.

- <Claim 51>

  A computer program element for centralizing system-wide communication management and control within filter rules comprising: a computer readable medium; first program instructions for providing filter statements syntax for accepting parameters in the form of a selector, each selector specifying selector field, a logical operator, and a set of values (Fiveash, figure 4); second program instructions for executing filtering by constructing and evaluating logical expressions of arbitrary length, said logical expressions selectively including said logical operator selected from a set of logical operators, at least one said selector field, and at least one said value (Jackowski, column 15, lines 21-43; column 11, lines 41-46; and column 12, line 61 through column 13, line 7); said selector referencing data that does not exist in IP packets (Lucovsky, column 2, lines 13-35) including data obtained, for an inbound packet, by executing a look-ahead function within a protocol stack including an IP layer, a transport layer, a sockets layer, and an application layer and

which, for said IP inbound packet, said IP layer provides to said transport layer said inbound IP packet, marked as non-deliverable, and receives back from said transport layer indicia, provided to said transport layer by said sockets layer, identifying the application layer application to which said packet would have been delivered (Lucovsky, column 8, lines 23-32 and Jackowski, column 15, line 66 through column 16, line 32); ; and wherein said first and second program instructions are recorded on said computer readable medium.

Since the combination of Lucovsky and Fiveash discloses all of the above limitations, claims 16-21, 23, 38, 40, 46, and 51 are rejected.

Conclusion

21. The prior art made of record and not relied upon is considered pertinent to the applicant's disclosure.

- Wong et al. (U.S. Patent Number 5,835,727) disclosed a system for controlling access to services within a computer network that maintains a profile of filtering rules.
- Chopra et al. (U.S. Patent Number 6,510,509) disclosed a high-speed rule processing system that maintains compare engines that include memory for storing instructions and operands, arithmetic logic for performing comparisons, and control circuitry for interpreting instructions and operands.
- Xie et al. (U.S. Patent Number 6,772,347) disclosed a firewall engine that includes a set of rules for sorting incoming IP packets into initially allowed packets and initially denied packets.

22. The applicant's amendment necessitated the new grounds of rejection presented in this office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). The applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

23. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Victor Lesniewski whose telephone number is 571-272-3987. The examiner can normally be reached on Monday through Thursday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Bunjob Jaroenchonwanit can be reached on 571-272-3913. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

VL

Victor Lesniewski
Patent Examiner
Group Art Unit 2152

BUNJOB JAROENCHONWANIT
PRIMARY EXAMINER